

Biased correction of SPP: Linear and Quantile Mapping Technique

Presented by:

Begum Rabeya Rushi

Regional Associate,

NASA/SERVIR-HKH

Earth System Science Center | University of Alabama, Huntsville.

Steps

- Making all cell size 0.05 and extent similar
- Biased correction of CHIRPS using gauge data
- Biased correction of SPP using CHIRPS

Table 1. SPPs Used in This Study and Their Specifications

SPP	Provider	Primary Sensor Type	Spatial Resolution	Temporal Resolution	Spatial Coverage	Temporal Coverage
PERSIANN-CCS ^a [Hong et al., 2004]	UCI	Infrared	0.04°×0.04°	3 hourly	37.8°N–40.6°S 28.0°W–56.2°E	2006–2010
CMORPH ^b [Joyce et al., 2004]	NOAA-CPC	Infrared + Passive Microwave	0.25°×0.25°	3 hourly	60°N–60°S 180°E–180°W	1998 to near present
TMPA-RT ^c [Huffman et al., 2007]	NASA	Visual + Infrared + Passive Microwave + Active Microwave	0.25°×0.25°	3 hourly	50°N–50°S 180°E–180°W	1998 to near present
CHIRPS ^d [Funk et al., 2014]	UCSB	Merged Products + In-situ precipitation observations	0.05°×0.05°	daily	50°N–50°S 180°E–180°W	1981 to near present

^aPrecipitation Estimation from Remotely Sensed Information using Artificial Neural Networks-Cloud Classification System.

^bClimate Prediction Center Morphing Technique.

^cTropical Rainfall Measuring Mission (TRMM) Multisatellite Precipitation Analysis - Real Time.

^dClimate Hazards Group InfraRed Precipitation with Station data.

- 3 Hourly to Daily Conversion

Roy et al. 2016

Linear Method Equation

$$P_{\text{cor},m,d} = P_{\text{raw},m,d} \times \frac{\mu(P_{\text{obs},m})}{\mu(P_{\text{raw},m})},$$

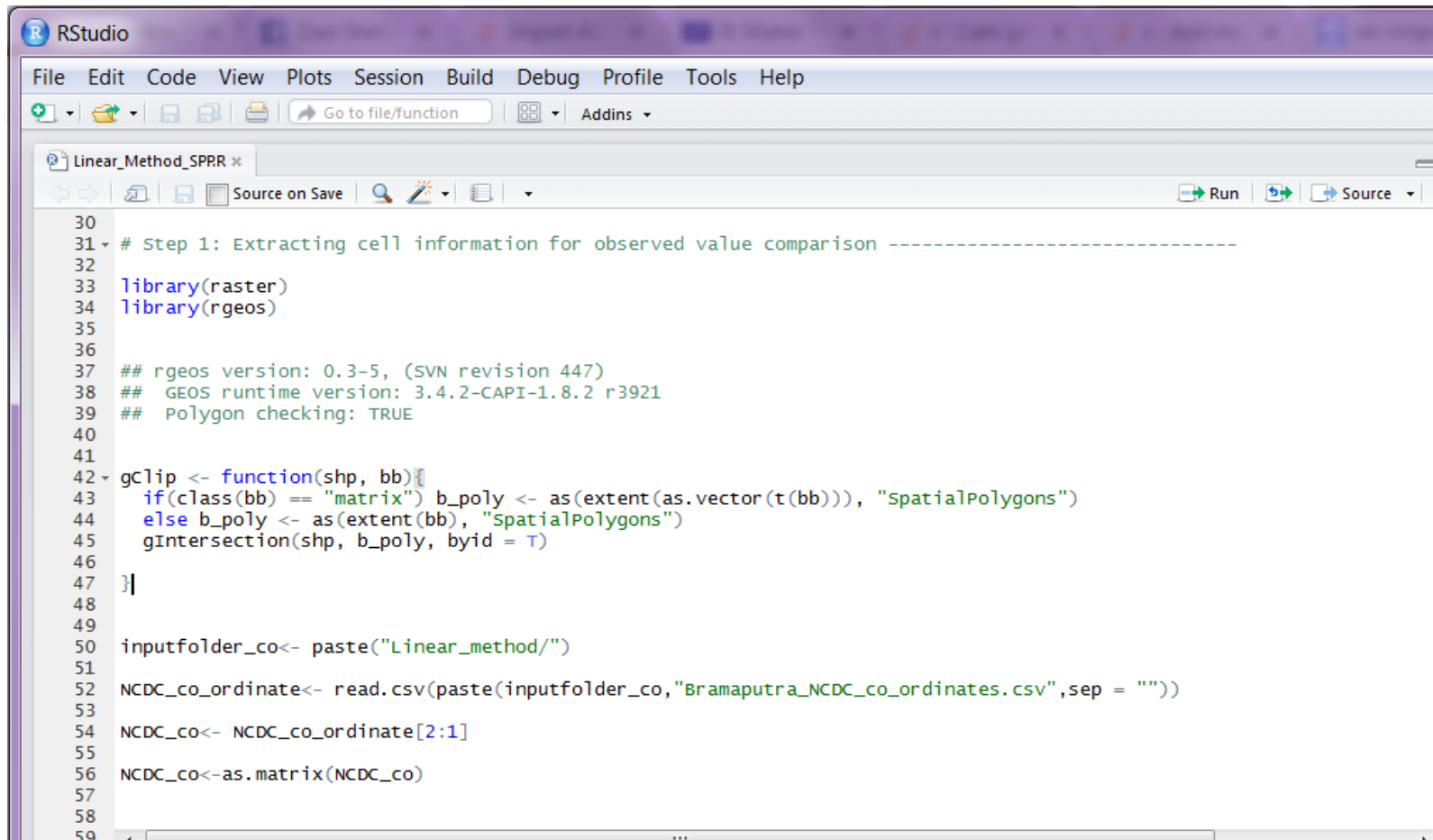
where $P_{\text{cor},m,d}$ is corrected precipitation at given month m on d th day.

$P_{\text{raw},m,d}$ is satellite precipitation at given month m on d th day. $\mu(P_{\text{obs},m})$ mean value of CHIRPS precipitation at given month m and $\mu(P_{\text{raw},m})$ mean value of satellite precipitation at given month m

Linear Method: Mean Calculation

- For entire time period, we will get 12 different mean values for each month; for example from 2005 to 2012, we will get 12 mean precipitation raster/matrix for Jan-Dec.
- Each cell will have 12 mean values for each month. If CHIRPS raster is 30x60; for each month, we have to calculate a 30x60 raster.
- We will have 12 raster files for each SPP. Extent and pixel size should be same as CHIRPS.
- Final output would be 30x60 raster

Correction of Extracting Gauge Station Information Area for the River Basin:



```
30
31 # Step 1: Extracting cell information for observed value comparison -----
32
33 library(raster)
34 library(rgeos)
35
36
37 ## rgeos version: 0.3-5, (SVN revision 447)
38 ## GEOS runtime version: 3.4.2-CAPI-1.8.2 r3921
39 ## Polygon checking: TRUE
40
41
42 gclip <- function(shp, bb){
43   if(class(bb) == "matrix") b_poly <- as(extent(as.vector(t(bb))), "SpatialPolygons")
44   else b_poly <- as(extent(bb), "SpatialPolygons")
45   gIntersection(shp, b_poly, byid = T)
46 }
47
48
49
50 inputfolder_co<- paste("Linear_method/")
51
52 NCDC_co_ordinate<- read.csv(paste(inputfolder_co,"Bramaputra_NCDC_co_ordinates.csv",sep = ""))
53
54 NCDC_co<- NCDC_co_ordinate[2:1]
55
56 NCDC_co<-as.matrix(NCDC_co)
57
58
59
```

Area Extent and Time Extent

```
61
62
63
64
65 Year<- c("2015")
66
67 Basin<- c("Brahmaputra")
68 ##"Ganges", "Indus", "Meghna")
69
70
71
72 inputfolder<- paste(Year,"/chirps-v2.0.",sep = "")
73 if(!file.exists(inputfolder))dir.create(inputfolder)
74
75 ## first days of years
76 date1<- seq(as.Date("2015-1-1"), as.Date("2015-12-31"), "days")
77 date_seq<- gsub("-", ".",date1)
78 head(date_seq)
79
80 date_seq<- as.matrix(date_seq)
81
82 no_days<-length(date_seq)
83
```

Step 2: Extracting cell information for observed value comparison

```
crop_folder<-paste("chirps/test/Brahmaputra/chirps/cropped_chirps/",Year[y],"/",sep="")
if(!file.exists(crop_folder))dir.create(crop_folder)
crop_chirps_file<-paste(crop_folder,date_seq[d],".tif",sep="")
writeRaster(chirps_global_crop, filename=crop_chirps_file, format="GTiff", overwrite=TRUE)

# NCDC_point<-matrix(0,1,2)
# NCDC_point[,1]<-as.numeric(NCDC_co[p,3])
# NCDC_point[,2]<-as.numeric(NCDC_co[p,2])

st_chirps[,2:4]<-data.frame(coordinates(NCDC_co),extract(chirps_global_crop,NCDC_co))
```


Step 3: Mean Monthly Component for CHIRPS and Gauge Data

```
for (m in 1:length(mon)) {  
  myList <- list()  
  
  for (y in 1:length(Year)) {  
  
    inputfolder<- paste("C:/chirps/test/Brahmaputra/chirps_obs_daily_comparison/",Year[y],sep = "")  
    setwd(inputfolder)  
  
    files <- list.files(path=".",pattern = paste(Year[y],".",sprintf("%02d", mon[m]),".*",sep="")) #path=inputfolder,  
    for(i in files) { myList[[length(myList)+1]]<-as.matrix(read.csv(i)) }  
  
  }  
  
  MonthlyMean<-Reduce("+", myList)/length(myList)  
  
  mean_monthly<-as.data.frame(MonthlyMean)  
  
  mean_monthly$Ratio<- mean_monthly$obs_prdp/mean_monthly$chirps_prdp  
  
  mean_monthly_garbage_removal <- mean_monthly[!is.infinite(mean_monthly$Ratio),]  
  mean_monthly_final <- mean_monthly_garbage_removal[!is.na(mean_monthly_garbage_removal$Ratio),]
```

Chirps Correction

```
195
196 - for (y in 1:length(Year)) {
197
198 -   for (m in 1:length(mon)) {
199
200     inputfolder_chirps<- paste("C:/chirps/test/Brahmaputra/chirps/cropped_chirps/",Year[y],"/",sep = "")
201     setwd(inputfolder_chirps)
202
203
204
205     files_chirps <- list.files(path=".",pattern = paste(Year[y],".",sprintf("%02d", mon[m]),".*",sep="")) #path=inputf
206
207
208
209 -   for(i in files_chirps) {
210
211     require(raster)
212
213     directory<- paste(inputfolder_chirps,i,sep = "")
214
215     chirps_daily<- as.matrix(raster(directory))
216
217
218
219     monthly_factor<- as.numeric(lapply(paste(mon[m],"monthly_obs_bias_factor", sep = "_"),get))
220
221     corrected_chirps<-data.matrix(chirps_daily*monthly_factor,rownames.force = T)
222
```

Corrected CHIRPS and SPP: Mean Monthly Matrix

```
202 outputfolder<- paste(Year,"/",Basin[b],"/",sep = "")
203 if(!file.exists(outputfolder))dir.create(outputfolder)
204
205
206 outputfile<- paste(outputfolder,date_seq[d],".precip.chirps.",Basin[b],".txt",sep = "")
207 writeRaster(PRCrop25,outputfile,format="ascii",NAflag=-9999,overwrite=TRUE)
208
209 }
210
211 }
212
213
214 # Step 2: Load Data (CHIRPS and SPP) -----
215
216 data_names<-c("chirps","persian")
217
218 no_data<-c(1:2)
219
220 # Step 3: Make mean monthly matrix for chirps and SPP -----
221
222
223 for (d in 1:length(data_names)) {
224
225   for (m in 1:length(mon)){
226     myList <- list()
227     for (y in 1:length(years)) {
228
229
230 inputfolder<- paste("c:/chirps/test/Brahmaputra/",data_names[d],"/",years[y],"/",sep = "")
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

```
234
235 for(i in files) { myList[[length(myList)+1]]<-as.matrix(raster(i)) }
236
237
238
239 }
240 MonthlyMean<-Reduce("+", myList) / length(myList)
241 mean_monthly<-as.matrix(MonthlyMean)
242
243 rb <- raster(mean_monthly)
244 class(rb)
245
246 # replace with correct coordinates
247 extent(rb) <- c(82,98,23.75,31.5)
248
249
250 outputfile<- paste("c:/chirps/test/Brahmaputra/",data_names[d],"/",sprintf("%02d", mon[m]),".mean.monthly.chirps",sep = "")
251
252 writeRaster(rb,outputfile,format="ascii",NAflag=-9999,overwrite=TRUE)
253
254 }
255
256 }
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

270:1 Step 6: SPP Correction

Console C:/chirps/Linear method/

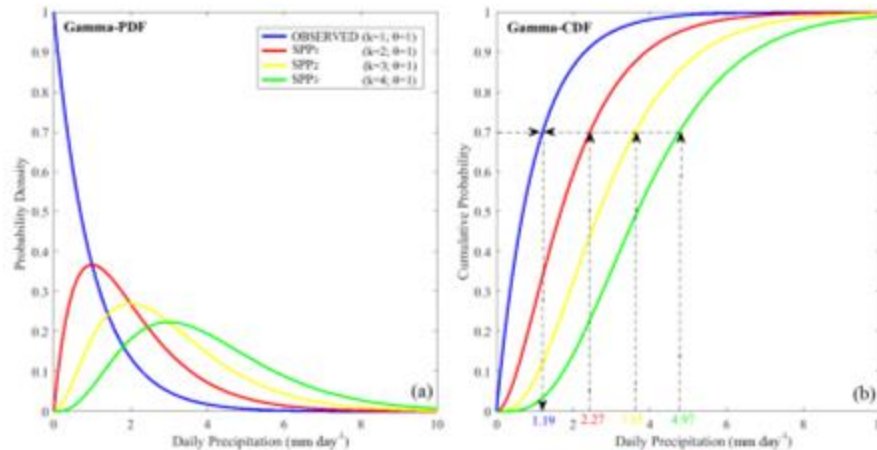
SPP Correction

```
monthly_chirps_factor<-raster(paste("C:/chirps/test/Brahmaputra/", "corrected_chirps", "/", mon[m], "_corrected_chirps_m\n", mon[m], "_persian_monthly_bias_factor.tif")\nmonthly_spp_factor<-raster(paste("C:/chirps/test/Brahmaputra/", "persian", "/", mon[m], "_persian_monthly_bias_factor.tif")\nmonthly_bias_factor<-overlay(monthly_chirps_factor, monthly_spp_factor, fun=function(x, y){ x/y} )\ncorrected_spp<-overlay(spp_daily, monthly_bias_factor, fun=function(x, y){ x*y})
```

Quantile Mapping

- Satellite and CHIRPS data covering the same period of record are used to create a "quantile map" of each population
- Gamma Probability Density Function (Gamma-PDF) is considered for precipitation distribution
- The Gamma-PDF is fitted for CHIRPS and satellite products at every grid-point and for all 12 months separately
- The Gamma Cumulative Distribution Function (Gamma-CDF) is used for determining probability associated with precipitation

Quantile Mapping



- a) The Gamma Probability Density Function (Gamma-PDF) is applied assuming different shapes (k parameter) for each dataset.
- b) The respective Gamma Cumulative Distribution Function (Gamma-CDF) for CHIRPS and SPPs is matched for a Probability ($P=0.7$) using the Inverse Gamma Function, which is finally used to calculate the bias-corrected daily satellite precipitation estimates.

[From Valdés-Pineda et al. \(2016\), Open Access Article in Hydrology and Earth System Sciences Discussions](#)

Loading Corrected CHIRPS and SPPs

```
29
30
31 for (p in 1:length(mon)){
32   myList <- list()
33   counter<-0
34   counter2<-0
35   for (y in 1:length(years)) {
36
37     # Loading CHIRPS and SPP -----
38
39     obsInputfolder<- paste("C:/chirps/quantile/Brahmaputra/chirps/",years[y],"/",sep = "")
40
41
42     satInputfolder<- paste("C:/chirps/quantile/Brahmaputra/persian/",years[y],"/",sep = "")
43
44
45
46
47     files_chirps <- list.files(path=obsInputfolder,pattern = paste("2015",".",sprintf("%02d", mon[p]),".*",sep="")) #path=
48
49     files_sat <- list.files(path=satInputfolder,pattern = paste("2015",".",sprintf("%02d", mon[p]),".*",sep="")) #path=inp
50
51
52
53
54 for(i in files_chirps) {
55   counter<-counter+1
56   print(counter)
57   obsDirectory<- paste(obsInputfolder,i,sep = "")
58   if (counter==1){
59
```

Making a 3-D Matrix for Each Month:

```
91
92 ▾ # Making 3-D Matrix for Each Month Considering All Years -----
93
94
95 dim(chirps) <- c(dim(raster(SatDirectory))[1], dim(raster(SatDirectory))[2], counter) # 3-D array formation
96
97 #dim(chirps) <- c(dim(raster(SatDirectory))[1], dim(raster(SatDirectory))[2], length(files_chirps)) # converting it to
98 Drizzle<-1 # less than 1 mm rain is considered drizzle
99 chirps[which(chirps<Drizzle)]<-0
100
101 chirps
102
103 dim(sat_prdp) <- c(dim(raster(paste(SatInputfolder,j,sep = "")))[1], dim(raster(paste(SatInputfolder,j,sep = "")))[2],
104 Drizzle<-1 # less than 1 mm rain is considered drizzle
105 sat_prdp[which(sat_prdp<Drizzle)]<-0
106
107 #CHIRPS<-array(3:63, dim=c(3,4,5))
108
109
110 #z <- array(1:60, dim=c(3,4,5))
111 #z[1,1,1]<-0 ## for testing
112
113 #GammaCDF <- array(0, dim=c(3,4,5))
114 #BCz<- array(0, dim=c(3,4,5)) # for storing bias corrected data
115
116 GammaCDF_chirps <- array(0, dim=c(dim(raster(obsDirectory))[1], dim(raster(obsDirectory))[2],counter))
117 GammaCDF_sat<- array(0, dim=c(dim(raster(SatDirectory))[1], dim(raster(SatDirectory))[2], counter2)) # for storing bia:
118
```


Parameter Identification of Gamma-PDF and Quantile Estimation

```
141
142 if (length(IndexNonZeroCHIRPS)>4 & length(IndexNonZeroSat)>4 & length(unique(NonZeroCHIRPS)) >4 & length(unique(Nonz
143 { ### at least more than 2 points are required for curve fitting and more than 3 unique values
144 #CHIRSParmsLambda[i,j]<-fitdistr(CHIRPS[i,j,], "gamma")$estimate[1] #lambda OR SHAPE
145 #CHIRSParmsTheta[i,j]<-fitdistr(CHIRPS[i,j,], "gamma")$estimate[2] #theta or rate
146 CHIRSParmsLambda[m,n]<-fitdistr(NonZeroCHIRPS, "gamma")$estimate[1] #lambda OR SHAPE
147 CHIRSParmsTheta[m,n]<-fitdistr(NonZeroCHIRPS, "gamma")$estimate[2] #theta or rate
148
149
150
151 #GammaParmsLambda[i,j]<-fitdistr(z[i,j,], "gamma")$estimate[1] #lambda
152 #GammaParmsTheta[i,j]<-fitdistr(z[i,j,], "gamma")$estimate[2] #theta
153 GammaParmsLambda[m,n]<-fitdistr(NonZeroSat, "gamma")$estimate[1] #lambda
154 GammaParmsTheta[m,n]<-fitdistr(NonZeroSat, "gamma")$estimate[2] #theta
155
156
157 #GammaCDF[i,j]<-pgamma(z[i,j,], GammaParmsLambda[i,j], rate = GammaParmsTheta[i,j], log = FALSE)
158 NonZeroGammaCDF<-pgamma(NonZeroSat, GammaParmsLambda[m,n], rate = GammaParmsTheta[m,n], log = FALSE)
159 #print(NonZeroGammaCDF)
160
161 #BCz[i,j,] <-qgamma(GammaCDF[i,j,],CHIRSParmsLambda[i,j], CHIRSParmsTheta[i,j])
162 GammaCDF_sat[m,n,IndexNonZeroSat]<-qgamma(NonZeroGammaCDF,CHIRSParmsLambda[m,n], CHIRSParmsTheta[m,n]) #inverse
163 }else {
164   print(NonZeroSat)
165   GammaCDF_sat[m,n,IndexNonZeroSat]<- NonZeroSat ## no bias correction is done if only 2 points are available
166
167
168 }
169
```

Acknowledgement:

- Professor Juan Valdes and members of his research group
- Emily, Africa and Lee